Evaluation Metrics

Jovan Pehcevski
INRIA Rocquencourt, France
jovan.pehcevski@inria.fr

Benjamin Piwowarski
Yahoo! Research Latin America
bpiwowar@yahoo-inc.com

SYNONYMS

Performance metrics; Evaluation of XML retrieval effectiveness.

DEFINITION

An *evaluation metric* is used to evaluate the effectiveness of information retrieval systems and to justify theoretical and/or pragmatical developments of these systems. It consists of a set of measures that follow a common underlying evaluation methodology.

There are many metrics that can be used to evaluate the effectiveness of semi-structured text (XML) retrieval systems. These metrics are based on different evaluation assumptions, incorporate different hypotheses of the expected user behaviour, and implement their own evaluation methodologies to handle the level of overlap among the XML information units.

HISTORICAL BACKGROUND

Compared to traditional information retrieval, where whole documents are the retrievable units, information retrieval from XML documents creates additional evaluation challenges. By exploiting the logical document structure, XML allows for more focused retrieval by identifying information units (or XML elements) as answers to user queries. Due to the underlying XML hierarchy, in addition to finding the most specific elements that at the same time exhaustively cover the user information need, an XML retrieval system needs to also determine the appropriate level of answer granularity to return to the user. The *overlap problem* of having multiple nested elements, each containing identical textual information, can have a huge impact on XML retrieval evaluation [7].

Traditional information retrieval evaluation measures (such as recall and precision) mostly assume that the relevance of an information unit (e.g. a document) is binary and independent of the relevance of other information units, and that the user has access to only one information unit at a time. Furthermore, they also assume that the information units are approximately equally sized.

These two assumptions do not hold in XML retrieval, where the information units are nested elements of very different sizes. As nested elements share parts of the same information, an evaluation metric for XML retrieval can no longer assume than the relevance of elements is independent. Moreover, since users can access different parts of an XML document, it also can no longer be assumed that they will have access to only one element at a time. Each of the evaluation metrics for XML retrieval supports the above assumptions to a different extent.

Another limitation of the traditional information retrieval metrics is that they are not adapted to the evaluation of specific retrieval tasks, which could use more advanced ways of presenting results that arise naturally when dealing with semi-structured documents. For example, one task in XML retrieval is to present the retrieved elements by their containing documents, which allows for the relevant information within each document to be identified more easily.

Over the past five years, the Initiative for the Evaluation of XML Retrieval (INEX) has been used as an arena to investigate the behaviour of a variety of evaluation metrics. Most of these metrics are extensions of traditional information retrieval metrics, namely precision-recall and cumulated gain.

Precision-recall is a bi-dimensional metric that captures the concentration and the number of relevant documents retrieved by an information retrieval system. An alternative definition of this metric calculates precision at a

Table 1: Common notations used to describe formulae of XML retrieval metrics. The column "Section" gives the section number where a more detailed description can be found (if any).

Notation	Section	Short description
e		An XML element
e_i		The ith XML element in the list
$spe(e)$	1.3	The (normalised) specificity
$exh(e)$	1.3	The (normalised) exhaustivity
$q(e)$	1.4	A quantization function
$\mathfrak{I}$	1.5	The set of ideal elements
$\mathcal{L}$	1.5	The ideal ranked list of elements
ℓ		Arbitrary recall level
size(e)		The size of element e, usually in number of characters
overlap(i)	1.6	The level of overlap between the ith element of the list and the previously returned elements

given recall level ℓ (between 0 and 100%) as the probability that a retrieved document is relevant, provided that a user wants to see ℓ percent of the relevant documents that exist for the topic of interest [13]. The precision-recall metric was the first one extended for the purposes of XML retrieval evaluation.
The Cumulated Gain (CG) metrics [3] rely on the idea that each retrieved document corresponds to a gain for the user, where the gain is being a value between 0 and 1. The metric then simply computes the CG at a given rank k as a sum of the gains for the documents retrieved between the first rank and the rank k. When normalised, the CG value is somewhat similar to recall, and it is also possible to construct an equivalent of precision for CG. The importance of extending this metric for XML Retrieval lies in the fact that it allows for non-binary relevance, which means it can capture elements of varying sizes and granularity.

SCIENTIFIC FUNDAMENTALS
Throughout this document, we use a common notation to describe the formulae of the different evaluation metrics for XML retrieval. The notation is presented in Table 1. We also assume that any XML element can be represented as a textual segment that spans the text corresponding to that XML element. This conceptual representation is practical, as it is possible to define the intersection, the union, the inclusion, and the size of any two segments.

1 Evaluation concepts

In XML retrieval, the commonly used ad hoc retrieval task simulates how a digital library is typically used, where information residing in a static set of XML documents is retrieved using a new set of topics. Different sub-tasks can be distinguished within the broad ad hoc retrieval task.

1.1 XML retrieval tasks

The main XML retrieval tasks, considered to be sub-tasks of the main INEX ad hoc retrieval task, are the following:

Thorough, where XML retrieval systems are required to estimate the relevance of a retrieved element and return a ranked list of all the overlapping relevant elements.

- Focused, where the returned ranked list consists of non-overlapping relevant elements.

- Relevant in Context (RiC), where systems are required to return a ranked list of relevant articles, where for each article a set of non-overlapping relevant elements needs to be correctly identified.

- Best in Context (BiC), where the systems are required to return a ranked list of relevant articles, where for each article the best entry point for starting to read the relevant information within the article needs to be correctly identified.

1.2 User behaviour

The evaluation metrics typically model a sequential user browsing behaviour: given a ranked list of answer elements, users start from the beginning of the list and inspect one element at a time, until either all the elements in the list have been inspected, or users had stopped inspecting the list since their information needs were fully satisfied. However, while inspecting a ranked list of elements, users of an XML retrieval system could also have an access to other structurally-related elements, or indeed could be able to inspect the *context* where the answer elements reside (which may be supported by features such as browsing, scrolling, or table of contents).
Accordingly, in addition to modelling the sequential user model, the evaluation metrics should also be able to model various user browsing behaviours.

1.3 Relevance dimensions

The relevance of a retrieved XML element to a query can be described in many ways. It is therefore necessary to define a relevance scale that can be used by the evaluation metrics. Traditional information retrieval usually uses a binary relevance scale, while in XML retrieval there is a multi-graded (or continuous) relevance scale that uses the following two relevance dimensions:

Exhaustivity (denoted exh), which shows the extent to which an XML element covers aspects of the information need.

- Specificity (denoted spe), which shows the extent to which an XML element is focused on the information need.

The two relevance dimensions have evolved over the years (readers are referred to the relevance definitional entry for more details). For simplicity, we will assume that each relevance dimension uses a continuous relevance scale with values between 0 and 1. For example, the four-graded relevance scale used by the two dimensions in INEX from 2002 until 2004 can be mapped onto the values 0, $\frac{1}{3}$, $\frac{2}{3}$ and 1.
We respectively denote $exh(e)$ and $spe(e)$ as the normalised Exhaustivity and Specificity of an XML element e. They can take values between 0 and 1.

1.4 Quantisation

Quantisation is the process of transforming the values obtained from the two relevance dimensions into a single normalised relevance score (which again takes values between 0 and 1). It is used to represent the extent to which the retrieved element is relevant.
For example, the strict quantisation function can be used to measure the XML retrieval performance when only highly relevant elements are targets of retrieval, while the generalised quantisation function can be used to measure the performance when elements with multiple degrees of relevance are targets of retrieval:

$$q_{\text{strict}}(e) = \begin{cases} 1 & \text{if } exh(e) = spe(e) = 1 \\ 0 & \text{otherwise} \end{cases} \qquad q_{\text{gen}}(e) = exh(e) \times spe(e)$$

The strict quantisation can therefore be used to reward systems that only retrieve elements that are fully exhaustive and specific, while the generalised quantisation rewards systems that retrieve elements with multiple relevance degrees.

1.5 Ideality

The concept of ideality emerged in XML retrieval as a concept that is used to distinguish those among all judged relevant elements that users would prefer to see as answers. For example, in order to distinguish between the intrinsic relevance of a paragraph from the inherited relevance of its containing section, we could say that, even though both elements are relevant, only the paragraph is *ideal*. By definition, an ideal element is always relevant but the reverse is true only in traditional information retrieval.
Ideal elements, unlike relevant elements, can be assumed to be independent. Note that this assumption is similar to the independence of document relevance in traditional information retrieval; that is, ideal elements, as documents, can overlap *conceptually* (they can contain same answers to the underlying information need) as long as they do not overlap *physically*. In XML retrieval, this assumption implies that ideal elements cannot be nested. Note that ideality can be extended to more general units than elements, namely the passages.

Construction of ideal sets and lists To construct a set $\mathfrak{I}$ of ideal elements, one has to make hypotheses about the underlying retrieval task and the expected user behaviour [6].
One example of methodology for identifying the ideal elements is as follows [4]: given any two elements on a relevant path, the element with the higher quantised score is first selected. A *relevant path* is a path in the document tree that starts from the document element and ends with a relevant element that either does not contain other elements, or contains only irrelevant elements. If the two element scores are equal, the one deeper in the tree is chosen. The procedure is applied recursively to all overlapping pairs of elements along a relevant path until only one element remains. It is important that the methodology for identifying ideal elements closely reflects the expected user behaviour, since it has been shown that the choice of methodology can have a dramatic impact on XML retrieval evaluation [4].
Given a set of ideal elements, and an evaluation metric that uses that set, it is then possible to construct an ideal list $\mathcal{L}$ of retrieved elements that maximises the metric score at each rank cutoff.

1.6 Near misses and overlap

Support of near misses is an important aspect that needs to be considered by the evaluation metrics for XML retrieval. *Near misses* are elements close to an ideal element, which act as entry points leading to one or more ideal elements. It is generally admitted that systems that retrieve near misses should be rewarded by the evaluation metrics, but in lesser extent than when ideal elements are retrieved [5].
Early attempts that extended the traditional information retrieval metrics to support XML retrieval rewarded near misses by assigning partial scores to the elements nearby an ideal one [2, 7]. However, this implies that systems that return only ideal elements will never achieve a 100% recall, since both ideal elements and near misses have to be returned to achieve this level of recall [11].
Moreover, these metric extensions are commonly considered to be "overlap positive" [14], which means that they reward systems for retrieving twice the same ideal element, either directly or indirectly, and that the total reward for retrieving that ideal element increases with the number of times it is retrieved. To cater for this problem, overlap neutral and/or negative evaluation metrics have since been developed [2, 6].
It is therefore important to be able to compute the degree of overlap between an element e_i and other elements previously retrieved in the ranked list $(e_1, \ldots, e_i)$. A commonly adopted measure is the percentage of text in common between the element and the other previously retrieved elements:

$$\text{overlap}(i) = \frac{\text{size}\left(e_i \cap \bigcup_{j=1}^{i-1} e_j\right)}{\text{size}\,(e_i)}$$

The overlap function equals 0 when there is no overlap between the element e_i and any of the previously retrieved elements e_j, and equals 1 when there is full overlap (i.e. either all its descendants or one of its ancestors have been retrieved). A value between 0 and 1 denotes intermediate possibilities.

2 Metric properties

An evaluation metric for XML retrieval should provide a support for the following properties.

Faithfulness: the metric should measure what it is supposed to measure (*fidelity*) and it should be *reliable* enough so that its evaluation results can be trusted.

- *Interpretation*: the outcome of the evaluation metric should be easy to interpret.
- *Recall/Precision*: the metric should capture both *recall* and *precision*, as they are complementary dimensions whose importance have been recognised in traditional information retrieval (some retrieval tasks put more focus on recall while others prefer precision).
- *Ideality*: the metric should support the notion of ideal elements.
- *Near misses*: the metric should be able to properly handle near misses.
- *Overlap*: the metric should properly handle the overlap among retrieved and judged elements.
- *Ideality graded scale*: the metric should be able to support multi-graded or continuous scales, in order to distinguish the ideality of two elements.

Table 2: Metric properties, and the extent to which each of the XML retrieval metrics supports them. In the table, "y" stands for yes, "n" for no, "i" for indirect, "+" for overlap positive, "-" for overlap negative, and "=" for overlap neutral. The question mark "?" signifies unclear or not demonstrated property.

Property \ Metric	inex_eval	inex_eval_ng	nXCG	ep/gr	T2I	GR	PRUM	EPRUM	HiXEval
Research publication	[2]	[2]	[6]	[6]	[1]	[11]	[12]	[10]	[8]
INEX metric (years)	02-04	03	05-06	05-06				06	07
Faithfulness	?	?	y	y	?	y	y	y	y
Interpretation	n$^{(a)}$	n$^{(a)}$	y$^{(b)}$	y$^{(b)}$	y	y	y	y	y$^{(b)}$
Recall	y	y	y	y	y	y	y	y	y
Precision	y	y	n	y	y	y	y	y	y
Near misses	i	i	y	y	y	y	y	y	y
Overlap	+	-	=	=	=	=	=	=	=
Ideality	n	n	y	y	y	y	y	y	y$^{(c)}$
Ideality graded scale	n/a	n/a	y	y	n	y	n	y	y$^{(d)}$
Explicit user model	n	n	n	n	y	y	y	y	n
XML Retrieval Tasks									
Thorough	y	y	y	y	?	y	y	y	y
Focused	?	?	y	y	y	y	y	y	y
RiC	i	i	i	i	i	y	y	y	i
BiC	i	i	i	i	i	y	y	y	i

(a) but for some special cases
(b) with parameter α set to 0 or 1
(c) highlighted passages are the ideal units in the case of HiXEval
(d) the ideality of an element is fixed and directly proportional to the amount of highlighted text

- *User models and retrieval tasks*: the metric should be able to model different *user behaviours* and support different *retrieval tasks*, since XML retrieval systems support a variety of features that allow information access.

Table 2 summarises the above metric properties and provides an overview of the extent to which each of the evaluation metrics for XML retrieval (described in the next section) provides a support for them.

3 Evaluation metrics

In this section, we present the different evaluation metrics that were proposed so far in XML retrieval.

3.1 The inex_eval metric

For three years since 2002, the inex_eval metric [2] has been used as the official INEX metric to evaluate the effectiveness of XML retrieval systems. This metric supports *weak ordering* of elements in the answer list [13], where one or more elements are assigned identical retrieval status values by an XML retrieval system. For simplicity, we restrict our discussion to the case where elements are fully ordered.
The inex_eval metric assumes that the degree (or probability) of relevance of an element e is directly given by the quantisation function $q(e)$. Its degree of non relevance can be symmetrically defined as $(1 - q(e_i))$. At a given rank k, it is then possible to define the expected number of relevant (resp. non relevant) $R(k)$ (resp. $I(k)$) elements as follows:

$$R(k) = \sum_{i \leq k} q(e_i) \quad I(k) = \sum_{i \leq k} (1 - q(e_i))$$

For a given recall level ℓ, the *Precall* [13] metric estimates the probability that a retrieved element is relevant to a topic (assuming that a user wants to find ℓ% of the relevant elements in the collection, or equivalently $\ell \cdot N$ relevant elements). If k_ℓ is the smallest rank k for which $R(k)$ is greater or equal to $\ell \cdot N$, then precision is defined as follows:

$$\text{Precision}(\ell) = \frac{\text{number of seen relevant units}}{\text{expected search length}} = \frac{\ell \cdot N}{k_\ell} \quad (1)$$

where N is assumed to be the expectation of the total number of relevant elements that can be found for an INEX topic, i.e. $N = \sum_e q(e)$ across all the elements of the collection.
Beyond the lack of support for various XML retrieval tasks, the main weakness of the `inex_eval` metric is that one has to choose (with the quantisation function) whether the metric should allow near misses or should be overlap neutral – both are not possible. To support overlap, it is possible to compute a set of ideal elements by setting the normalised quantisation scores of non-ideal elements to 0, thus not rewarding near misses. To reward near misses, the quantisation function should give a non-zero values for elements nearby the ideal elements, but then the system will get fully rewarded only if it returns both the ideal and the other relevant elements. Another problematic issue is the use of non-binary relevance values inside the `inex_eval` formula shown in equation (1), which makes the metric ill-defined from a theoretical point of view.

3.2 The `inex_eval_ng` metric

The `inex_eval_ng` metric was proposed as an alternative evaluation metric at INEX 2003 [2]. Here, the two relevance dimensions, *Exhaustivity* and *Specificity*, are interpreted within an *ideal concept space*, and each of the two dimensions is considered separately while calculating recall and precision scores. There are two variants of this metric, which differ depending on whether overlap among retrieved elements is penalised or not: `inex_eval_ng(o)`, which penalises overlap among retrieved elements; and `inex_eval_ng(s)`, which allows overlap among retrieved elements. Unlike the `inex_eval` metric, this metric directly incorporates element sizes in their relevance definitions.
With `inex_eval_ng(o)`, precision and recall at rank k are calculated as follows:

$$\text{Precision}(k) = \frac{\sum_{i=1}^{k} spe(e_i)\cdot \text{size}(e_i)\cdot(1-\text{overlap}(i))}{\sum_{i=1}^{k} \text{size}(e_i)\cdot(1-\text{overlap}(i))} \quad \text{Recall}(k) = \frac{\sum_{i=1}^{k} exh(e_i)\cdot(1-\text{overlap}(i))}{\sum_{i=1}^{N} exh(e_i)} \quad (2)$$

With `inex_eval_ng(s)`, recall and precision are calculated in the same way as above, except that here the overlap function is replaced by the constant 0 (by which overlap among the retrieved elements is not penalised).
The `inex_eval_ng` metric has an advantage over `inex_eval`, namely the fact that it is possible to penalise overlap. However, due to the fact that it ignores the ideality concept, the metric has been shown to be very unstable if one changes the order of elements in the list, in particular the order of two nested elements [12]. Moreover, `inex_eval_ng` treats the two relevance dimensions in isolation by producing separate evaluation scores, which is of particular concern in evaluation scenarios where combinations of values from the two relevance dimensions are needed to reliably determine the preferable retrieval elements.

3.3 The `XCG` metrics

In 2005 and 2006, the eXtended Cumulated Gain (XCG) metrics [6] were adopted as official INEX metrics. The XCG metrics are extensions of the cumulated gain metrics initially used in document retrieval [3].

Gain and overlap When the cumulated gain (CG) based metrics are applied to XML retrieval, they follow the assumption that the user will read the whole retrieved element, and not any of its preceding or following elements. An element is partially seen if one or more of its descendants have already been retrieved ($0 < \text{overlap}(i) < 1$), while it is completely seen if any of its ancestors have been retrieved ($\text{overlap}(i) = 1$).
To consider the level of overlap among judged relevant elements, the XCG metrics makes use of an ideal set of elements (see Section 1.5), also known as the *ideal recall base*. To consider the level of overlap among the retrieved elements in the answer list, the XCG metrics implement the following result-list dependent relevance value (or gain) function:

$$\text{gain}(i) = \begin{cases} q(e_i) & \text{if overlap}(i) = 0 \\ (1-\alpha) \cdot q(e_i) & \text{if overlap}(i) = 1 \\ \alpha \cdot \frac{\sum_{j/e_j \subseteq e_i} \text{gain}(j) \cdot \text{size}(e_j)}{\text{size}(e_i)} + (1-\alpha) \cdot q(e_i) & \text{otherwise} \end{cases}$$

The parameter α influences the extent to which the level of overlap among the retrieved elements is considered. For example, with α set to 1 (Focused task), the gain function returns 0 for a previously fully seen element, reflecting the fact that an overlapping (and thus redundant) element does not bring any retrieval value in evaluation. Conversely, the level of overlap among the retrieved elements is ignored with α set to 0 (Thorough task).
The gain formula cannot guarantee that the sum of the gain values obtained for descendants of an ideal element are smaller than the ideal element gain, and so it is necessary to "normalise" the gain value by forcing an upper gain bound [6].

`XCG` metrics Given a ranked list of elements for an INEX topic, the cumulated gain at rank k, denoted as $XCG(k)$, is computed as the sum of the normalised element gain values up to and including that rank:

$$XCG(k) = \sum_{i=1}^{k} \text{gain}(i) \tag{3}$$

Two XCG metrics used as official XML retrieval metrics at INEX in 2005 and 2006 are `nXCG` and `ep/gr`. The `nXCG` metric is a normalised version of $XCG(k)$, defined as the ratio between the gain values obtained for the evaluated list to the gain values obtained for the ideal list.
The `ep/gr` metric was defined as an extension of `nXCG` in order to average performances over runs and to define an equivalent of precision. It consists of two measures: effort-precision ep, and gain-recall gr.
The gain-recall gr, calculated at the rank k, is defined as:

$$gr[k] = \frac{XCG[k]}{\sum_{x \in \mathcal{I}} \text{gain}(x)} \tag{4}$$

The effort-precision ep is defined as the amount of relative effort (measured as the number of visited ranks) a user is required to spend compared to the effort they could have spent while inspecting an optimal ranking. It is calculated at a cumulated gain level achieved at rank k and is defined as:

$$ep[k] = \frac{\min\{i | XCG_{\mathcal{L}}(i) \geq XCG(k)\}}{k} \tag{5}$$

where the indice $\mathcal{L}$ means that the score is evaluated with respect to an ideal list of relevant elements. An ep score of 1 reflects an ideal performance, in which case the user made the minimum necessary effort (computed in number of ranks) to reach that particular cumulated gain. An ep/gr curve can then be computed by taking pairs $(gr[k], ep[k])$ for varying rank k values.
The XCG metrics have advantages over `inex_eval` and `inex_eval_ng`, since its use of an ideal list ensures that the metric is overlap neutral. It also properly handles near misses by the use of an appropriate quantisation function. However, the construction of the ideal set of elements relies on heuristics [4]. Other problems of the metric is that the gain is difficult to interpret for values of α other than 0 or 1, which also makes the outcome of the metric somewhat difficult to interpret.

3.4 The `T2I` metric

The tolerance to irrelevance (T2I) metric [1] relies on the same evaluation assumptions as `inex_eval`, but includes a different user model more suited to semi-structured documents. The underlying user model is based on the intuition that a user processes the retrieved list of elements until their tolerance to irrelevance have been reached (or until they found a relevant element), at which point the user proceeds to the next system result. The T2I metric has only been theoretically proposed, and is yet to be implemented and evaluated.

3.5 The (E)PRUM and GR metrics

The Expected Precision Recall with User Modelling (EPRUM) metric [10], which was used as an alternative evaluation metric at INEX in 2005 and as one of the official ones in 2006, extends the traditional definitions of precision and recall to model a variety of user behaviours. EPRUM is unique among all the INEX metrics in that it stochastically defines the user browsing behaviour. It is the last defined within a set of three metrics, the previous one being GR (Generalised recall) and PRUM (Precision Recall with User Modelling).

The user model From a retrieved element, the user can navigate using the corpus structure. The *context* of a list item is defined as the set of elements that can be reached through navigation from it. This includes the pointed elements but also the context of the pointed elements (siblings, ancestors, etc.). To model the user behaviour inside the context, the three metrics rely on a set of probabilities on simple events of the form "navigating from a list item to an element in the corpus". The probabilities of navigating from rank j in the list to an element x can be set to values estimated by any adequate method and is denoted $\mathrm{P}(j \rightsquigarrow x)$. When a user is over with this exploration, they *consult* the next entry of the list and repeat the process until their information needs are satisfied. Note that this user model is general enough so as to cope with all INEX tasks, since there is no constraint on how a rank is defined.

Users *see* an element when they navigate to it from another element or from the list, and they *discover* an element if they see it for the first time. The distinction between "seen" and "discovered" is important because the system is rewarded only when elements are discovered. The probability that the user discovers f ideal elements when consulting ranks between 1 and k included is then given by:

$$\mathrm{P}(F_k = f) = \sum_{\substack{A \subseteq \mathfrak{I} \\ |A|=f}} \prod_{x \in A} \mathrm{P}(x \in \mathcal{S}_k) \prod_{x \in \mathfrak{I} \setminus A} \mathrm{P}(x \notin \mathcal{S}_k) \tag{6}$$

where $\mathcal{S}_k$ is the set of all elements seen by the user who has consulted ranks 1 to k. The probability that an element was seen is computed with $\mathrm{P}(x \in \mathcal{S}_k) = 1 - \prod_{j=1}^{k} (1 - \mathrm{P}(j \rightsquigarrow x))$.

GR, PRUM and EPRUM The GR metric is a generalisation of recall with the above specified user model. It simply estimates the expected number of discovered elements at a given rank k, and divides it by the expected number of ideal elements in the database in order to get a normalised value. PRUM is defined as the probability that a consulted list item leads the user to discover an ideal element. Its most important limitation is that it does not handle well non-binary assessments.

EPRUM defines precision based on the comparison of two minimum values: the minimum rank that achieves the specified recall over *all* the possible lists and over the *evaluated* list. For a given recall level ℓ, precision is thus defined as the percentage of effort (in minimum number of consulted ranks) a user would have to make when consulting an ideal list with respect to the effort when consulting the evaluated list:

$$\text{Precision}(\ell) \quad = \quad \mathbb{E}\left[\frac{\begin{array}{c}\text{Minimum number of consulted}\\ \text{list items for achieving a recall } \ell \text{ over all lists}\end{array}}{\begin{array}{c}\text{Minimum number of consulted list items}\\ \text{for achieving a recall } \ell \text{ over the evaluated list}\end{array}}\right] \tag{7}$$

where the assumption is that when the user cannot achieve recall ℓ in the evaluated list, then the minimum number of consulted list items for the evaluated list is infinite (this assumption is the same as in traditional information retrieval). This measure is an extension of the standard precision-recall metric.

It is similarly possible to extend the traditional definition of precision at a given rank k. If the expected recall of the evaluated list at rank k is r_k, then precision at rank k is defined as the ratio of the minimum number of consulted list items over all possible lists to achieve recall r_k to the number of consulted ranks k.

The EPRUM metric solved some problems of PRUM and substantially reduced its complexity. It also allowed to properly handle graded ideality. The EPRUM metric advantages are the fact that it handles all the INEX tasks through its user model parameters, that the user model is very flexible (for example allowing to reward near misses that are not direct ancestors or descendant of an ideal element), and that the outcome of the metric can easily be interpreted. However, like the XCG metrics, it assumes that the ideal set of elements and the ideal list of retrieved elements can easily be determined, which is shown to be not as straightforward in XML retrieval [4].

3.6 The HiXEval metric

Since 2005, a highlighting assessment procedure is used at INEX to gather relevance assessments for the XML retrieval topics. In this procedure, assessors from the participating groups are asked to highlight sentences representing the relevant information in a pooled set of retrieved documents. To measure the extent to which an XML retrieval system returns relevant information, INEX started to employ evaluation metrics based on the HiXEval metric [8, 9]. This is motivated by the need to directly exploit the INEX highlighting assessment procedure, and it also leads to evaluation metrics that are natural extensions of the well-established metrics used in traditional information retrieval.

HiXEval only considers the Specificity relevance dimension, and it credits systems for retrieving elements that contain as much highlighted (relevant) text as possible, without also containing a substantial amount of non-relevant text. So, instead of counting the number of relevant elements retrieved, HiXEval measures the amount of relevant text retrieved. Like the XCG metrics, it makes the assumption that the user will read the whole retrieved element, and not any of its preceding or following elements. An element is partially seen by the user if one or more of its descendants have already been retrieved, while it is completely seen if any of its ancestors have been retrieved.

Let $\text{rsize}(e_i)$ be the amount of highlighted (relevant) text contained by an element e retrieved at rank i, so that if there is no highlighted text in the element, $\text{rsize}(e_i) = 0$.[1] To measure the value of retrieving relevant text from e_i, the relevance value function $\text{rval}(i)$ is defined as follows:

$$\text{rval}(i) = \begin{cases} \text{rsize}(e_i) & \text{if overlap}(i) = 0 \\ (1-\alpha) \cdot \text{rsize}(e_i) & \text{if overlap}(i) = 1 \\ \text{rsize}(e_i) - \alpha \cdot \sum\limits_{j/e_j \subseteq e_i} \text{rval}(j) & \text{otherwise} \end{cases}$$

As with the XCG metrics, the parameter α is a weighting factor that represents the importance of retrieving non-overlapping elements in the ranked list.

Precision and recall at a rank k are defined as follows:

$$\text{Precision}(k) = \frac{\sum\limits_{i=1}^{k} \text{rval}(i)}{\sum\limits_{i=1}^{k} \text{size}(e_i)} \quad \text{Recall}(k) = \frac{1}{Trel} \cdot \sum_{i=1}^{k} \text{rval}(i) \tag{8}$$

In the above equation, $Trel$ represents the total amount of highlighted relevant text for an INEX topic. Depending on the XML retrieval task, different $Trel$ values are used by the metric. For example, for the Focused task $Trel$ is the total number of highlighted characters across all *documents*. This means that the total amount of highlighted relevant text for the topic represents the sum of the sizes of the (non-overlapping) highlighted passages contained by all the relevant documents. Conversely, for the Thorough task $Trel$ is the total number of highlighted characters across all *elements*. For this task, the total amount of highlighted relevant text for the topic represents the sum of the sizes of the (overlapping) highlighted passages contained by all the relevant elements.

The precision and recall scores can be combined in a single score using the standard F-measure (their harmonic mean). By comparing the F-measure scores obtained from different XML retrieval systems, it would be possible to see which system is more capable of retrieving as much relevant information as possible, without also retrieving a substantial amount of non-relevant information.

HiXEval has the advantage of using a naturally defined ideal unit, namely a highlighted passage, and thus overcomes the problem of defining a set of ideal elements in an arbitrary way. One shortcoming of this metric is that it makes the assumption that the degree of ideality of a passage is directly proportional to the passage size. It also shares the same issue identified with the XCG metrics that, with α values different from 0 or 1, the interpretation of the output of the $\text{rval}(i)$ function is not very straightforward.

KEY APPLICATIONS

[1] Note that $\text{rsize}(e_i)$ can also be represented as: $\text{rsize}(e_i) = spe(e_i) \cdot \text{size}(e_i)$.

Web search

Due to the increasing adoption of XML on the World Wide Web, information retrieval from XML document collections has the potential to be used in many Web application scenarios. Accurate and reliable evaluation of XML retrieval effectiveness is very important for improving the usability of Web search, especially if the evaluation captures the extent to which XML retrieval can be adapted to a particular retrieval task or a user model. This could certainly justify the increasing usage of XML in the ever-growing number of interactive Web search systems.

Digital Libraries

Reliable evaluation of XML retrieval effectiveness is also important for improving information retrieval from digital libraries, especially since there is a large amount of semi-structured (XML) information that is increasingly stored in modern digital libraries.

URL TO CODE

EvalJ project: http://evalj.sourceforge.net

CROSS REFERENCE

XML retrieval; Evaluation initiative for XML retrieval (INEX).

RECOMMENDED READING

[1] A.P. de Vries, G. Kazai, and M. Lalmas. Tolerance to irrelevance: A user-effort evaluation of retrieval systems without predefined retrieval unit. In *Proceedings of RIAO 2004*, pages 463–473, Avignon, France, 2004.

[2] N. Gövert, N. Fuhr, M. Lalmas, and G. Kazai. Evaluating the effectiveness of content-oriented XML retrieval methods. *Information Retrieval*, 9(6):699–722, 2006.

[3] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[4] G. Kazai. Choosing an ideal recall-base for the evaluation of the Focused task: Sensitivity analysis of the XCG evaluation measures. In *Comparative Evaluation of XML Information Retrieval Systems: Fifth Workshop of the INitiative for the Evaluation of XML Retrieval, INEX 2006*, volume 4518, pages 35–44, 2007.

[5] G. Kazai and M. Lalmas. Notes on what to measure in INEX. In *Proceedings of the INEX 2005 Workshop on Element Retrieval Methodology*, pages 22–38, Glasgow, UK, 2005.

[6] G. Kazai and M. Lalmas. eXtended Cumulated Gain measures for the evaluation of content-oriented XML retrieval. *ACM Transactions on Information Systems*, 24(4):503–542, 2006.

[7] G. Kazai, M. Lalmas, and A. P. de Vries. The overlap problem in content-oriented XML retrieval evaluation. In *Proceedings of the ACM-SIGIR International Conference on Research and Development in Information Retrieval*, pages 72–79, Sheffield, UK, 2004.

[8] J. Pehcevski. *Evaluation of Effective XML Information Retrieval*. PhD thesis, RMIT University, Melbourne, Australia, 2006.

[9] J. Pehcevski and J. A. Thom. HiXEval: Highlighting XML retrieval evaluation. In *Advances in XML Information Retrieval and Evaluation: Fourth Workshop of the INitiative for the Evaluation of XML Retrieval, INEX 2005*, volume 3977, pages 43–57, 2006.

[10] B. Piwowarski and G. Dupret. Evaluation in (XML) information retrieval: Expected precision-recall with user modelling (EPRUM). In *Proceedings of the ACM-SIGIR International Conference on Research and Development in Information Retrieval*, pages 260–267, Seattle, USA, 2006.

[11] B. Piwowarski and P. Gallinari. Expected ratio of relevant units: A measure for structured information retrieval. In *INEX 2003 Workshop Proceedings*, pages 158–166, 2003.

[12] B. Piwowarski, P. Gallinari, and G. Dupret. Precision recall with user modelling (PRUM): Application to structured information retrieval. *ACM Transactions on Information Systems*, 25(1):1–37, 2007.

[13] V. Raghavan, P. Bollmann, and G. Jung. A critical investigation of recall and precision. *ACM Transactions on Information Systems*, 7(3):205–229, 1989.

[14] A. Woodley and S. Geva. XCG overlap at INEX 2004. In *INEX 2005 Workshop Pre-Proceedings*, pages 25–39, 2005.